Flipping the Controls Classroom Around a MOOC

Jean-Pierre de la Croix and Magnus Egerstedt

Abstract— **Bridging the theory-practice gap in controls education is a well-known challenge. In this paper, we discuss how one can bridge this gap using a flipped classroom. Based on the recent MOOC (Massive Open Online Course), Control of Mobile Robots, we flipped the classroom in a senior robotics and controls class at the Georgia Institute of Technology. The students participated in the MOOC and came to class prepared to solve controls problems on robots. Key to this experience was not only the delivery of theoretical concepts via the MOOC, but also a hardware/software platform that provided a learning environment where exploratory, practical tinkering was grounded in solid theory. This paper reports on the findings of the flipped classroom experiment, as well as discusses why this classroom format is ideal for controls courses.**

I. Introduction

In the Spring of 2013, we tried an educational experiment by introducing a *flipped classroom* [1] at the Georgia Institute of Technology in the senior course ECE 4555: Embedded and Hybrid Control Systems [2]. In particular, the students learned the theoretical aspects of the material through the MOOC (Massive Open Online Course) Control of Mobile Robots [3], taught by the second author, and then came to class prepared to apply the material to simulation exercises as well as on real robotic hardware. In this paper, we report on the findings associated with this experiment, including discussions of the flipped classroom, what tools are needed to support such an endeavor, how one can structure a MOOC for upper-level engineering courses, and what the potential implications are for future controls curricula.

To set the stage for this educational experiment, it is paramount that we first mention a few words concerning the use of mobile robots in the controls curriculum at Georgia Tech. A few years ago, we included a robotics project in the introductory undergraduate controls course in the School of Electrical and Computer Engineering, with the first author as teaching assistant and the second author as instructor. Apart from the standard material involving Laplace transforms, transfer functions, and Bode plots, the students had to design controllers that would drive differential drive robots through a sequence of way-points in the shortest amount of time by carefully tuning the gains of a proportional-integral-derivative (PID) controller, as shown in Figure 1. And this modest modification to our "classic" controls course was a resounding success! The students were suddenly much more excited and engaged in class and, as controls educators, it was absolutely wonderful to hear groups of students argue about the relative merits of high I-gains. The subsequent year, we decided to increase the robotics content in the course by having the students design complete navigation systems that would make the robots negotiate cluttered environments in a safe and effective manner. But this incarnation ended up being just as much of a failure as the previous incarnation had been a success!

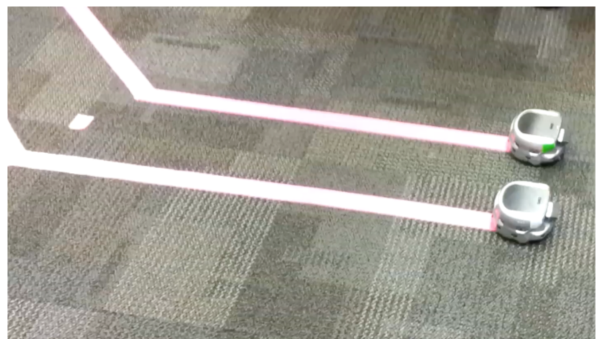

Fig. 1: A robotics competition as part of the introductory controls course at Georgia Tech.

What seemed to be at play was that the students stopped being systematic about their control design choices, and instead ended up hacking together complex and cumbersome solutions, which forced us to make one of two choices: either we revert to the previous, less hands-on robotics project or we escalate the focus on the robotics projects. We made the latter choice, which brings us to the flipped classroom [1]. We really wanted some way of making the robotics projects more elaborate and exciting, yet did not want to compromise on the technical content of the class. As such, the MOOC format provided a venue through which the technical content of the course could be outsourced.

Key to delivering such a flipped classroom is not only that the theoretical part can be outsourced, but also that an appropriate learning infrastructure is present that allows the instructors to bridge the two gaps that one always encounters when teaching controls and robotics classes, namely

1) *The theory-to-practice-gap:* How can the theoretical developments discussed in the "standard" controls classroom be mapped onto a physical platform in a way that supports the educational experience?
2) *The simulation-to-hardware-gap:* Just because something works in simulation, it certainly does not follow that it will work on actual hardware. How can this be remedied in a way that does not involve hours of parameter tuning?

This work was supported by an educational grant from Mathworks.

Jean-Pierre de la Croix and Magnus Egerstedt are with the School of Electrical and Computer Engineering, Georgia Institute of Technology, 777 Atlantic Drive NW, Atlanta, GA 30332-0250, USA {jdelacroix,magnus}@gatech.edu

Our solution to this problem was to develop a simulator, Sim.I.am, that lets students test robotic controllers both in simulation and in practice in a seamless manner.

The outline of this paper is as follows: In Section II, we briefly discuss the flipped classroom mechanics and how ECE4555 (the on-campus class) and the MOOC coexisted side-by-side during the Spring of 2013. In Section III, we then discuss the MOOC in more detail, as well as describe how it was designed explicitly with the flipped on-campus classroom in mind. This discussion is followed by a description of the simulator, Sim.I.am, in Section IV, and the paper ends with Section V, with a summary of the findings and a discussion of the lessons learned.

II. COURSE MECHANICS

ECE4555 Embedded and Hybrid Control Systems is a senior elective undergraduate course at Georgia Tech, running for 15 weeks, with students taking the course typically having completed at least one course on Signals and Systems (ECE3085) and one on Control Systems (ECE3550). With this in mind, ECE4555 fills the role of the first "advanced" controls class, with a tilt towards state space methods and control applications.

In this section, we discuss how we structured the flipped Spring 2013 course. The obstruction to overcome was one of timing, where a 7 week MOOC had to synchronize with a 15 week on-campus class. In fact, ECE4555 started a full three weeks prior to the start of the MOOC on Coursera. The MOOC also ended 5 weeks before the end of the semester in early May 2013. Therefore, it is useful to split the the discussion of the course mechanics into three parts: before the MOOC, during the MOOC, and after the MOOC.

A. Before the MOOC

The semester started three weeks prior to the start of the MOOC, which introduced an interesting problem: What should the students of the flipped classroom do before the MOOC starts? The first week was focused on the usual discussion of the syllabus, what students can expect from the course, as well as, a review of basic control theory and an introduction to robotics. We also posted a number of disclaimers that the course would not be a regular course and that there was a real chance that the workload would be significantly greater than a "normal" 3-credit course. Despite these warnings all 35 enrolled students decided to stay. To get the students familiar with the simulator and hardware platforms (Khepera III mobile robots), the second and third weeks included a detailed discussion of Sim.I.am and the robots, with the expectation that students would be well-prepared for the upcoming hands-on, in-class projects. As a result, the first three weeks were spent preparing the students for the upcoming seven weeks of synchronized MOOC lectures and hands-on projects.

B. During the MOOC

The MOOC started late January 2013 and lasted for seven week. Each week of the MOOC included lectures, multiple-choice homework, and corresponding hands-on projects the following week for the students of the flipped classroom. The projects were offset by a week to give the students a chance to digest the material, design controllers, test these controllers in the simulator before deploying their controllers on the real robots. And, to pass the on-campus, flipped class, students had to not only get a passing grade in the MOOC, but also complete the robotics projects.

The class met twice a week for 1.5h, and the first meeting time of the week was split into two halves. The first half was spent reviewing key concepts from the MOOC, while the other half was allocated to students working on designing and testing controllers in the simulator in small groups (these small groups were a function of the number of robots available, i.e. nine robots for 35 students). The second class of the week did not convene in the regular classroom, but rather in our research lab. We chose this location since it provided easy access to robots and enough space for all 35 students. As part of these in-class projects, the student groups had to complete six carefully designed projects:

1) *Go-to-Angle*: The students needed to write a controller for the robot to turn to a specific angle θ_d using a PID controller. This project also required the students to implement odometry for estimating the robot's pose (x, y, θ) and a transformation of the linear and angular velocities (v, ω) of the unicycle to the right and left wheel speeds (v_r, v_l) of the differential-drive robot.
2) *Go-to-Goal*: The students were asked to implement an improved version of the previous controller that would steer the robot towards a goal and drive the robot forward with a linear velocity inversely proportional to the distance to the goal.
3) *Avoid-Obstacles*: The third project required students to convert the raw infra-red (IR) sensor measurements to meaningful data and use this information to prevent the robot from driving into any obstacles. Students learned to filter the noisy IR sensor measurements.
4) *Switching and Blending*: The fourth project combined the go-to-goal and avoid-obstacles controllers by either hard switching between or blending the two controllers. This project introduced students to hybrid control, where a supervisor makes decisions about which controller to use based on specific events (e.g., an obstacle is detected within some critical distance).
5) *Follow-Wall*: The fifth project challenged students to create a controller that could follow an obstacle (or wall) to either the right or left of the robot. This project required students to implement a sliding-mode controller that was induced by the two control modes go-to-goal and avoid-obstacles.
6) *Navigation*: The final project built on all of the previous projects by combining all controllers in a supervisory controller in the form of a finite state machine (FSM). The purpose of the FSM was to solve the navigation problem of driving the robot from point A to point B without colliding with any of the obstacles. The environment was cluttered with both convex and

more problematic, non-convex obstacles.

These projects were structured such that each project would enforce particular, key control theory concepts, and the students would receive a handout with the instructions of what they needed to demonstrate by the end of class. There were typically three to four tasks that needed to be signed off by an instructor.

Before students were allowed to start on a task with the robots, they had to demonstrate that their controllers worked in the simulator. If the controller did not work in simulation, it would most likely not work on the real robot. As such, the student were expected to bring controllers to class that worked properly in simulation to maximize the amount of time that could be spent on deploying the controllers on the real robots. As instructors, we spent most of our time circling among all of the groups to ensure that we answered any questions that they had and check off on their progress as they demonstrated the successful operation of their controllers.

C. *After the MOOC*

Once the MOOC concluded, the semester was four weeks away from conclusion. These last weeks were dedicated to student-defined, open-ended projects. The only requirements for the final projects were that they had to be more involved than the previous six projects, and that it was unique amongst all of the groups. They were allowed to use any number of robots in their projects, as well as a motion capture system as a pseudo-indoor GPS system for the robots. The groups chose topics, such as simultaneous localization and mapping (SLAM), rapidly-exploring random trees (RRT) for navigation, and multi-agent formation control. They added these algorithms on top of Sim.I.am and demonstrated their finished projects at the end of the semester with a short presentation and demo.

Before we can properly discuss how this educational experiment fared, we first have to discuss, in some detail, both the MOOC and the simulator Sim.I.am, since they both played important roles in the flipped classroom.

III. Controls for the Masses

Although this paper is foremost about the flipped classroom experiment, it is necessary to discuss the MOOC itself at some length. The reason being that online resources for higher education have received a significant amount of attention during the last year, but upper-level engineering classes have been virtually absent from all the major MOOC content providers (Udacity, Coursera, edX). As such, it is not entirely clear how one should structure such a MOOC.

The first observation is that producing a MOOC takes a lot of time. The general wisdom is that an online course takes three times as much time to produce as a regular class.The second is that it requires a significant level of planning and organization since by the time the MOOC goes live, all the material is already produced. The outline of the MOOC Control of Mobile Robots, which ran for seven weeks, ended up being as follows:

- Week 1: Introduction to Controls
- Week 2: Mobile Robots
- Week 3: Linear Systems
- Week 4: Control Design
- Week 5: Hybrid Systems
- Week 6: The Navigation Problem
- Week 7: Putting It All Together.

Each week was broken down into 8 sub-lectures. Online education research has found that people cannot concentrate for more than 11 minutes or so, and all MOOC providers are following this model of breaking down the material into smaller segments [7]. As an example, Week 4 featured several short lectures focused on control design topics, such as pole-placement and controllability. These lectures were recorded during the Fall of 2012, and each lecture consisted of a video feed of the professor superimposed over lecture slides that could be annotated in real-time using a tablet, as shown in Figure 2. The lectures were supplemented by robotic simulations, experiments, and demonstrations.

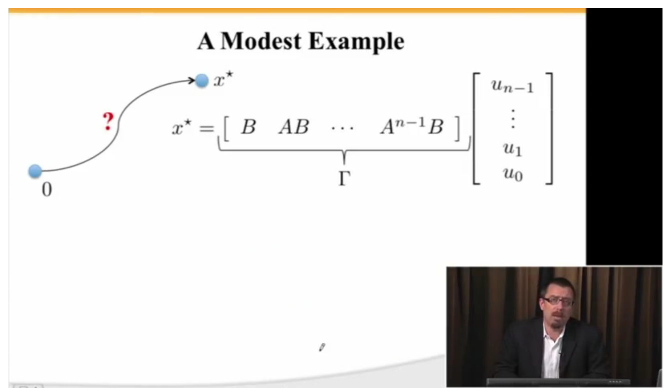

Fig. 2: Lectures were presented in this format.

A third observation is that it is impossible to grade 40,000 homework submissions individually. The remedy is automatic grading of multiple-choice questions, and not only do you have to produce interesting questions with good answers–you have to produce multiple interesting and good (albeit incorrect) answers. An example of such a multiple-choice question is shown below. The production of good assignments is key to creating a meaningful learning experience as the students spend a lot of time thinking about the questions and forming study-groups–physical *and* virtual.

The class went live at the end of January 2013 with over 40,000 registered students. This number is somewhat misleading, since a lot of people sign up for a MOOC and then never participate. In fact, the number of "active" participants during the first week (i.e., students that had watched a video, taken a quiz, or participated in the discussion forums) was closer to 12,000. The course quickly settled down at 9,000 active participants, with around 6,000 taking the weekly quizzes. A particularly pleasant surprise was that the students would help each other out on the discussion forums, start study groups, and find good online resources. At the end of March, the MOOC ended and around 4,000 certificates were handed out, which is well above the typical 6% retention rate [10]. To receive a certificate, a student needed to get 60% or higher on the quizzes, and to receive "Distinction", 90% or higher was needed. Almost half of all students that received certificates received them with distinction.

IV. SIM.I.AM

Key to bridging the theory-practice gap is an educational infrastructure that allows for high-fidelity control design as well as a mapping of control code both onto simulated as well as actual systems. To this end, we designed Sim.I.am, a mobile robot simulator [4]. This simulator enables students to design controllers for a mobile robot, test these controllers in a simulator, and then deploy the controllers on an actual robotic platform: the Khepera III mobile robot [5].

The Khepera III mobile robot (and its virtual analog in the simulator) is a differential-drive mobile platform with distance sensing IR sensors, wheel encoders, and WiFi connectivity. The simulator allows students to use the IR sensors and wheel encoders as feedback in their controllers, and control the mobile robot via input signals to the left and right wheels of the robot. The key feature of the simulator is that it allow students to design and implement controllers in MATLAB, test these in simulation, and then deploy it on the real robot without ever having to implement code outside of MATLAB (see Figure 3). The focus stays on the design of the controllers instead of implementation details that often derail the learning experience.

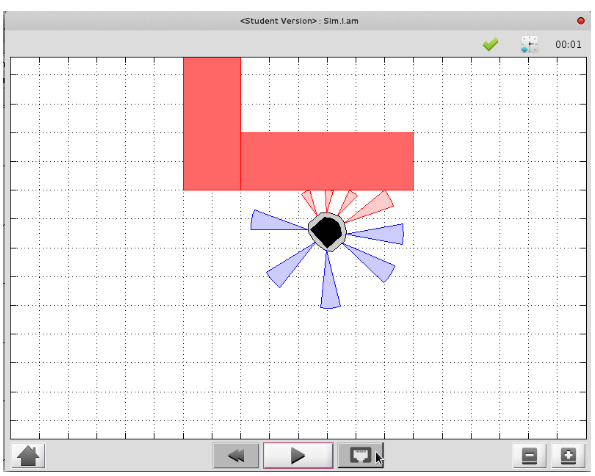

Fig. 3: Sim.I.am with the hardware connectivity button that allows for the transparent switching between the simulated robot or a real Khepera III mobile robot.

Sim.I.am provided the core of the experience for students enrolled in ECE4555. As part of the flipped classroom experience, students spent the beginning of the week learning a control theory concept through the MOOC course. For example, students would learn the mathematical formulation of a go-to-goal controller, which can drive a differential-drive robot from point A to point B. The first step required students to implement the go-to-goal controller in the simulator. While the simulator is a somewhat idealized version of the real world, it provides the students with a sufficient tool to test whether their controllers are behaving correctly. If a controller did not work in the simulator, it almost assuredly would not work on the real robot. The second step required students to deploy their go-to-goal controller on a real mobile robot. Rather than port their controller from MATLAB to C, the simulator provides a network interface (TCP/IP) that simply links the inputs/outputs from the students controllers to the real robot instead of the simulated robot. This approach allows students to focus their attention on adapting their control design to the real robot, rather than worry about porting their controller to C. Using the simulator in this way proved to be particularly successful; students progressed from running very simple controllers on the robots to implementing SLAM (Simultaneous Localization and Mapping) algorithms on the (virtual and real) robots within the span of a single semester (see Figure 4).

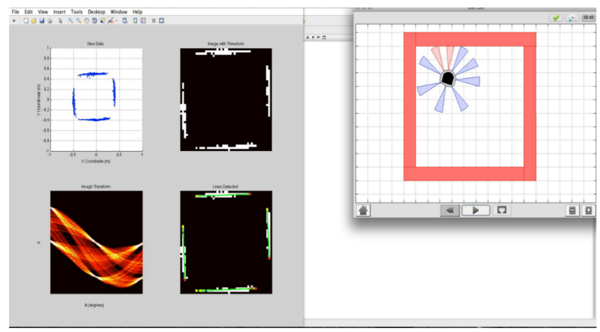

Fig. 4: SLAM algorithm added to Sim.I.am by ECE4555 students for their final project.

The simulator provided a similar experience to the students participating in the Coursera MOOC, Control of Mobile Robots. Students were provided with weekly assignments that mirrored the assignments in ECE4555; however, the option to run their controllers on an actual (not simulated) Khepera III mobile robot was not included. For students that did not have access to MATLAB, a stand-alone version of the simulator was included every week. This version of the simulator was compiled using the MATLAB Compiler, which did not allow students to implement their own controllers, but instead provided an XML file for modifying the parameters of already implemented controllers.

Sim.I.am included an instruction booklet that described the students's tasks for the week and provided details to guide the students in the implementation of their controllers and algorithms. For example, the task for the third week was for students to implement the different parts of a PID regulator that steers the mobile robot successfully to some goal location. This is known as the go-to-goal behavior. The three implementable subtasks were as follows:

1) Calculate the heading (angle), θ_g, to the goal location (x_g, y_g). Let u be the vector from the robot located at (x, y) to the goal located at (x_g, y_g), then θ_g is the angle u makes with the x-axis (positive θ_g is in the counterclockwise direction).

2) Calculate the error between θ_g and the current heading of the robot, θ.
3) Calculate the proportional, integral, and derivative terms for the PID regulator that compute ω.

Student were instructed to implement this functionality in `+simiam/+controller/GoToGoal.m`. Such `.m` files are part of MATLAB and MATLAB is the primary programming language of Sim.I.am. The simulator was implemented in MATLAB, because engineering students typically already understand how to code in MATLAB. This choice eliminated the need to spend precious class time to introduce students to an unfamiliar programming language. However, it is important to note that `GoToGoal.m` is not a script or function, but a class, because Sim.I.am is designed using the object-oriented programming features of MATLAB. Engineering students are typically not familiar with object-oriented design; however, it was not difficult to overcome this learning curve with a few pointers. For example, each controller should be thought of as its own object, because it is a self contained block with inputs and outputs. Also, a controller has its own memory. For example, it remembers the previous error, which is integral to computing the discrete time derivative in a PID controller. Treating controllers as objects also became more important later on as students had to reuse controller in order to combine them for more complex navigation behaviors.

In `GoToGoal.m`, students computed the proper linear and angular velocity needed to steer and drive the robot towards the goal location (x_g, y_g). The students tested this several times in the simulator for different pairs of (x_g, y_g) to verify that it worked. To help the student with this task, the simulator not only shows the robot moving around in its environment, but plotting tools are also provided to check whether the controller is steering the robot in the right directions. Testing this controller on the real robot is simply a matter of flipping a switch in the simulator, such that the control signals are routed to the real robot rather than its virtual counterpart. In a span of a week, students successfully transformed a control theory concept learned in the MOOC into a controller that drives a real robot from point A to point B. Sim.I.am made this classroom experience possible.

V. Outcomes

A. Student Surveys

To gauge the efficacy of the flipped classroom, we conducted a survey of how students viewed their experiences with the flipped classroom. We asked students to rank ten statements on a Likert scale (see [6]) from 1 to 5, i.e., from "completely disagree" to "completely agree":

1) I would like to see more flipped classrooms at GT.
2) The simulator and robotics projects helped me bridge the theory-practice gap.
3) I used what I learned in the online class when doing the robotics projects.
4) The use of a simulator and robots helped enforce concepts from the online class.
5) The use of a simulator made the robotics projects easier to complete.
6) The projects were of an appropriate level of difficulty.
7) The robotics projects were well-supported by the online material.
8) The robotics projects were well-supported by the simulator.
9) Open-ended final projects are a good idea.
10) I would recommend ECE4555 to other students.

We received 20 of the 35 surveys back with the statistics on the above statements from the survey compiled in Table I. Overall the experience was viewed as quite positive. Statistical significance is not something that can be gauged from this type of survey; however, there were two statements which scored below 4: *I used what I learned in the online class when doing the robotics projects* and *The robotics project were well-supported by the online material*. This observation is not surprising given the MOOC's focus on theory over practice. Most of the material about implementation and testing related to the projects was presented in class rather than the MOOC. However, overall these survey responses seem to indicate that the students viewed the flipped classroom as a success.

TABLE I: Survey statistics

Survey Question	Mean Response	Survey Question	Mean Response
1	4.40	6	4.45
2	4.50	7	3.95
3	3.85	8	4.60
4	4.30	9	4.45
5	4.60	10	4.80

B. General Flipped Classroom Observations

A flipped controls class is certainly not only doable, but in our experience can be quite successful. In ECE4555, the students managed to not only learn the material via the MOOC, but also apply it by building up robotics projects of increasing complexity. In fact, to pass the course, the students had to earn a certificate from the MOOC as well as complete all the projects. An example of such a project is shown in Figure 5. And, the projects and their respective application-oriented control design concepts (as listed in Section II-A) are not typically found in undergraduate controls courses. However, to make mobile robot control systems truly versatile and useful, they are all necessary building blocks.

As an important observation, the problem we encountered the year before, with students hacking together solutions, was completely absent from the flipped class. Instead, complex robotic behaviors were being designed in a systematic manner, using observer-based state feedback, hybrid automata, and linear-quadratic optimal control.

The one area where the flipped classroom was not entirely successful is the assessment. Since our explicit aim was to encourage the students to *apply* what they learned in class, we did not want the grades to be based on traditional, written exams. But, it is not entirely easy to establish who contributes to what in a group of four, nor is it easy to gauge the relative merits of different follow-wall behaviors. Future incarnations of this course will, as such, have to involve smaller group sizes, which in turn calls for more robots.

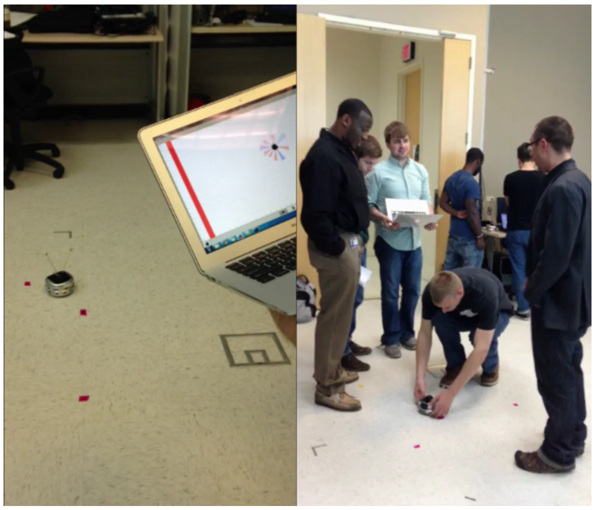

Fig. 5: A flipped classroom robotics project.

C. Lessons Learned about the MOOC

As controls MOOCs are virtually unknown, we also, for the sake of completeness, include a discussion of how the MOOC itself fared, decoupled from the on-campus, flipped classroom experience experience. During the first week of the MOOC, a small but vocal minority of the students were somewhat upset with the fact that they had not signed up for a "calculus class", or to quote one student on the forum: "Oh Bollocks! Matrices and integrals together!" At the end of the first week, the number of active participants was down to 12,000. Despite this, we decided to stick to our guns and not shy away from the math. And this seems to have been the right call. During the second week, we started receiving emails thanking us for not "dumbing down the course", and students were pointing out that finally all that linear algebra they took in school 20 years ago was paying off.

One of the added benefits from the MOOC was that it forced us to structure the on-campus, flipped class around appropriate chunks. It also reinforced the importance of having a clear and compelling arc through the material. But, for the MOOC, some aspects went well, while others did not go quite according to plan.

The Negatives:

- **40,000 is different from 40:** The amount of email traffic surrounding the MOOC is sometimes overwhelming, and the discussion forums are highly active.
- **An incredible time sink:** It takes significantly more time to prepare and run a MOOC than a standard class.
- **Engineering is hard:** Almost all MOOCs fall into one of three camps: The humanities, computer science, or "introduction to X". But there are virtually no upper-level engineering MOOCs. And there is a reason for this – engineering requires prerequisites, such as calculus, linear algebra, and Laplace transforms. Moreover, engineering courses are made better with hands-on labs, and there is simply no way meaningful, physical labs can be made a part of a free, online course given the current state of the technology.

Despite these negatives, the experience was overall very positive, as detailed below.

The Positives:

- **Appetite for serious engineering content:** The biggest and most surprising positive aspect of the controls MOOC is that there is strong demand for upper-level engineering content that does not skimp on the math.
- **Flipped classrooms:** Our experience flipping the classroom as Georgia Tech in conjunction with the MOOC worked out very well and we intend to follow this model in the future as well.
- **Incredibly rewarding:** A non-trivial aspect of teaching a large MOOC is that you are reaching people all over the world that you would otherwise have no chance of reaching. This democratizing aspect of the MOOC experience should not be understated.

As a final remark, we believe that controls curricula, which are by its very nature, theoretically focused, can benefit tremendously from the flipped classroom format. And, in this paper, we discussed one particular instantiation of this idea. Far from thinking of our experience as a definitive answer, we hope that it can be thought of as a starting point for more flipped controls classrooms to come.

References

[1] B. Tucker. The Flipped Classroom, *Education Next*, Vol. 12, No. 1, pp. 82-83, 2012.

[2] M. Egerstedt. ECE4555 Hybrid and Embedded Controls. Website accessed Sept. 2013. `http://http://users.ece.gatech.edu/~magnus/ece4555.html`.

[3] Coursera. Control of Mobile Robots. Website accessed Sept. 2013. `https://www.coursera.org/course/conrob`.

[4] J.P. de la Croix. Sim.I.am. Website accessed Sept. 2013. `http://gritslab.gatech.edu/projects/robot-simulator`.

[5] K-Team. Khepera III. Website accessed Sept. 2013. `http://www.k-team.com/mobile-robotics-products/khepera-iii`.

[6] Likert, Rensis. A Technique for the Measurement of Attitudes. Archives of Psychology 140: 155, 1932.

[7] S. Clemons. Brain-Based Learning: Possible Implications for Online Instruction, *International Journal of Instructional Technology and Distance Learning*, Vol. 2, No. 9, Sept. 2005.

[8] What Campus Leaders Need to Know About MOOCs, *EDUCAUSE Publications*, Dec. 2012.

[9] J. Gains. Will Free MOOCs Destroy Higher Education? *Digitopoly*, March 26, 2103.

[10] A. Watters. Dropping Out of MOOCs: Is It Really Okay? *Inside Higher Ed*, July 23, 2012.